Filtrage de contenu

Mise en oeuvre de serveurs mandataires (proxy)

Sommaire

- Introduction
- Principes généraux
  - Objectifs
  - Positionnement dans le SI
  - Limites
- Filtrage d'URL avec Squid
  - Installation
  - Paramétrage de base
  - Utilisation avancée
- Filtrage antivirus avec ClamaV
  - Installation
  - Paramétrage
  - Filtrage de courrier électronique

Introduction

- Le filtrage de contenu, également appelé filtrage applicatif, est un complément indispensable au seul filtrage réseau.
  - Les attaques sont en effet de plus en plus dirigées vers les applications
    - Cross Scripting, SQL injection, etc.
    - Phishing
    - Malwares
- Le filtrage applicatif répond donc à un besoin de plus en plus fort en matière de contrôle du contenu des flux qui transitent sur le réseau du SI.
- Les outils qui mettent en œuvre cette fonctionnalité sont très souvent des serveurs mandataires (proxy).
- Nous aborderons dans ce cours les cas du filtrage d'URL avec le proxy HTTP Squid et le filtrage antivirus avec le logiciel ClamAV.
- Par la suite, nous appellerons « serveur mandataire » ou « proxy » tout serveur qui joue le rôle d'intermédiaire entre des clients généralement internes et des serveurs généralement externes.

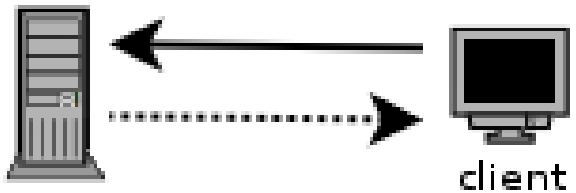
Principes généraux (1)

- Les outils de filtrage de contenu travaillent au niveau Application.
- Ils peuvent donc analyser ce qu'un pare feu classique de niveau 4 (Netfilter/Iptables) ne peut pas prendre en compte pour effectuer ses opérations de filtrage.
- L'association « filtrage niveau 4 + filtrage de contenu » accroît donc le niveau de sécurité et de contrôle des flux.
- Inconvénient : l'utilisation de ces outils est rarement transparente pour les utilisateurs.
  - Nécessite de modifier la configuration des postes de travail
    - Il est aussi possible d'utiliser des règles de redirection.

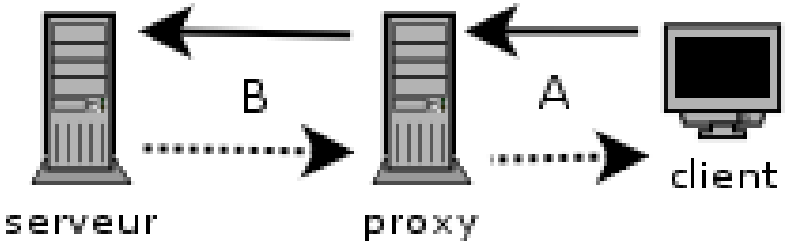
Principes généraux (2)

- Le proxy s'interpose entre le client et le serveur :

Relation Client/Serveur directe



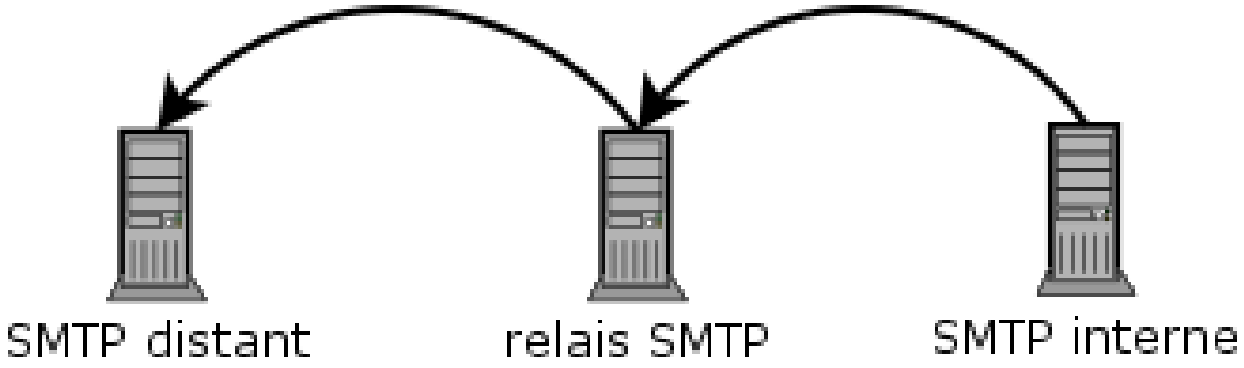
Relation Client/Serveur avec proxy



A. Le client dialogue avec le proxy qui prend la place du serveur.
B. La proxy prend la place du client vis-à-vis du serveur.

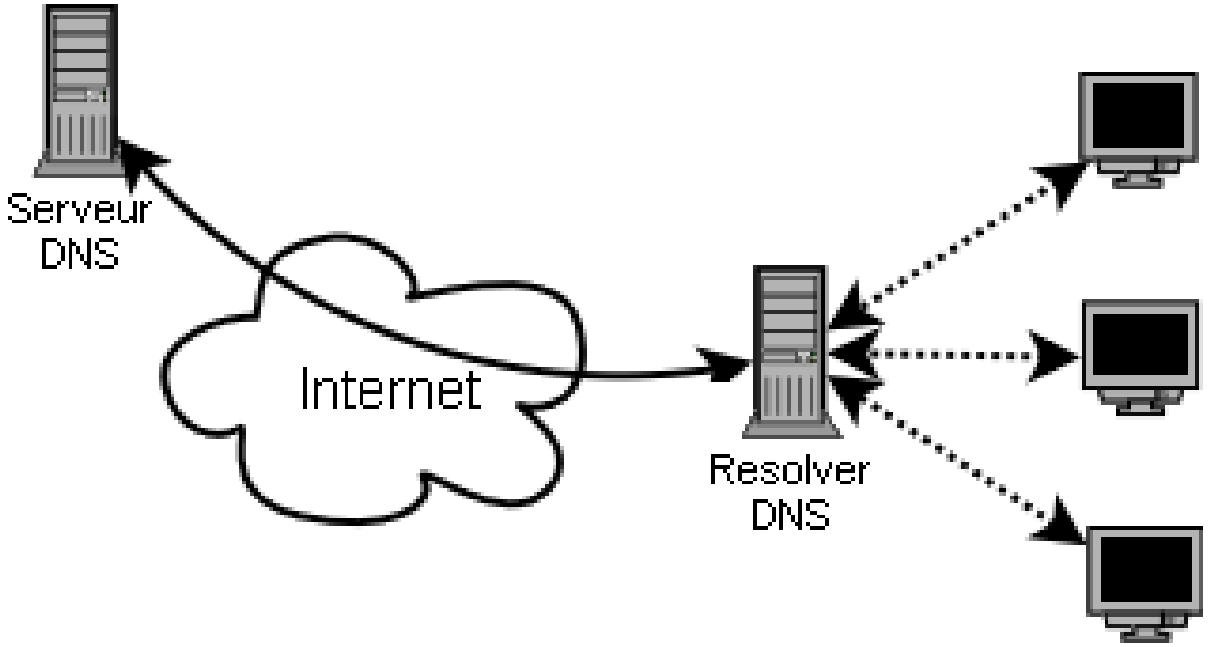
Principes généraux (3)

- Un relais SMTP peut être considéré comme un proxy :



Principes généraux (4)

- Dans une certaine mesure, un resolver DNS agit aussi comme un serveur mandataire :

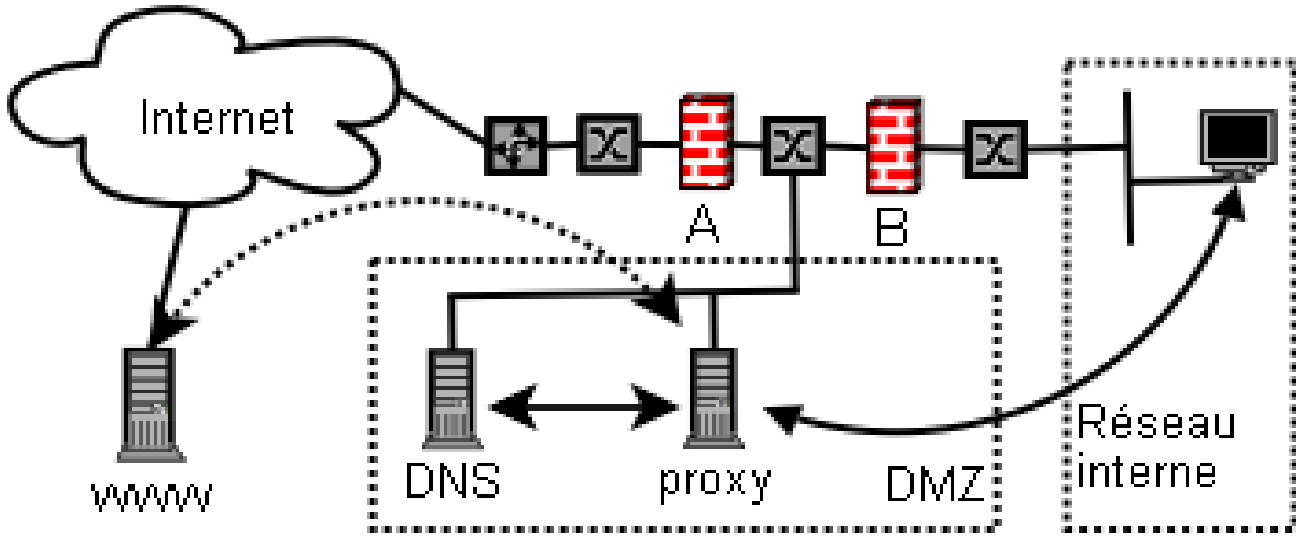


Objectifs

- Les proxies remplissent les fonctionnalités suivantes :
  - Ils participent à la mise en œuvre du cloisonnement de certaines zones.
    - Exemple des flux HTTP du réseau interne vers Internet que l'on oblige à transiter par un proxy en DMZ.
  - Ils augmentent les performances.
    - Mise en cache des pages HTML par exemple.
    - Mise en cache des réponses DNS
    - Déchargement du serveur SMTP interne pour les envois vers l'extérieur
  - Enfin, ils peuvent effectuer des opérations de contrôle (authentification) et de filtrage de contenu.

Positionnement

- Proxy pour flux Web :
  - Les postes clients internes sont configurés pour utiliser le proxy en DMZ et lui seul.
    - Les flux Web sont donc bloqués depuis le réseau interne vers Internet sur le pare feu B
  - Le proxy utilise les services d'un resolver DNS en DMZ pour la résolution.



Limites

- Tous les protocoles ne peuvent pas être proxifiés
  - Mais les principaux, heureusement, le sont :
    - HTTP, FTP, POP3, IMAP, SMTP
- Le filtrage de contenu se heurte, là aussi, au chiffrement de certains flux.
    - HTTPS
      - On conserve cependant la fonctionnalité d'authentification des clients.
- Le filtrage de contenu est une opération potentiellement gourmande en ressources machine. Plus le protocole est interactif, plus sa proxification peut s’avérer problématique.

Squid

Présentation

- Squid est un serveur mandataire (proxy) pour les protocoles HTTP, HTTPS et FTP. Il est utilisé comme proxy pour les navigateurs Internet.
- Quatre grandes fonctionnalités :
  - Cache de pages HTML et de fichier Web
  - Contrôle d'accès grâce à des ACLs.
  - Authentification des utilisateurs.
  - Traçage des usages du Web (journalisation des requêtes)

Contrôle d’accès

- Squid permet de contrôler les accès aux protocoles Web grâce à des ACL.
- Il est possible d’utiliser ces ACL pour déclencher une authentification des utilisateurs.

Filtrage d’URL

- Utilisation de SquidGuard comme redirecteur
- Permet de constituer des listes noires de domaines, de sites ou de mots-clefs interdits.
- Il est possible de télécharger des listes préétablies.

Configuration

- Installation
  - Sous Debian : apt-get install squid
- Généralement, un nouveau compte système et un nouveau groupe sont créés
  - Séparation des privilèges : Squid ne tourne pas sous l'identité root.
- Une fois installé, Squid utilise les objets suivants :
  - squid.conf : fichier de configuration du démon Squid
  - /var/lib/squid : répertoire du cache et des fichiers de log

Configuration

- Principaux paramètres de configuration du fichier squid.conf
- http_port : port sur lequel le démon squid est accessible.
  - Syntaxe : http_port [hostname:]port [[hostname:]port]
  - Valeur par défaut : 3128
  - Autre port usuel : 8080
  - Exemples :
    - http_port 3128
    - http_port 192.168.1.1:8080

Configuration

- Quand Squid est utilisé seul et n'appartient pas à une hiérarchie de cache, il faut désactiver les protocoles ICP et HTCP.
- ICP
  - Syntaxe : icp_port *port*
  - Valeur par défaut : 3130
  - Désactivation : icp_port 0
- HTCP
  - Syntaxe : htcp_port port
  - Valeur par défaut : 4827
  - Désactivation : htcp_port 0

Configuration

- Journalisation
  - log_fqdn on/off
- cache_access_log
  - Journalise les requêtes (HTTP) des clients
  - cache_access_log /path/to/access.log
- cache_log
  - Journalise l'activité du démon Squid
  - cache_log /path/to/squid.log
- cache_store_log
  - Journalise la gestion du cache par Squid
  - cache_store_log /path/to/store.log
- Autres
  - useragent_log, referer_log

Configuration

- Protocole FTP
- ftp_user
  - Identité qui sera présentée aux serveurs FTP anonymes
  - ftp_user squid@domain.ma
- ftp_list_width
  - Nombre de caractères affichés par ligne
- ftp_passive
  - on : le démon Squid se comporte en client FTP passif

Configuration

- DNS
- dns_timeout
  - Valeur du timeout. En cas de dépassement, un message d'erreur est retourné au client.
  - Par défaut : dns_timeout 5 minutes
- dns_nameservers
  - Par défaut, Squid utilise les serveurs DNS listés dans /etc/resolv.conf. Cette variable permet d'en utiliser d'autres.
  - dns_nameservers 192.168.1.10 192.168.2.10

Configuration

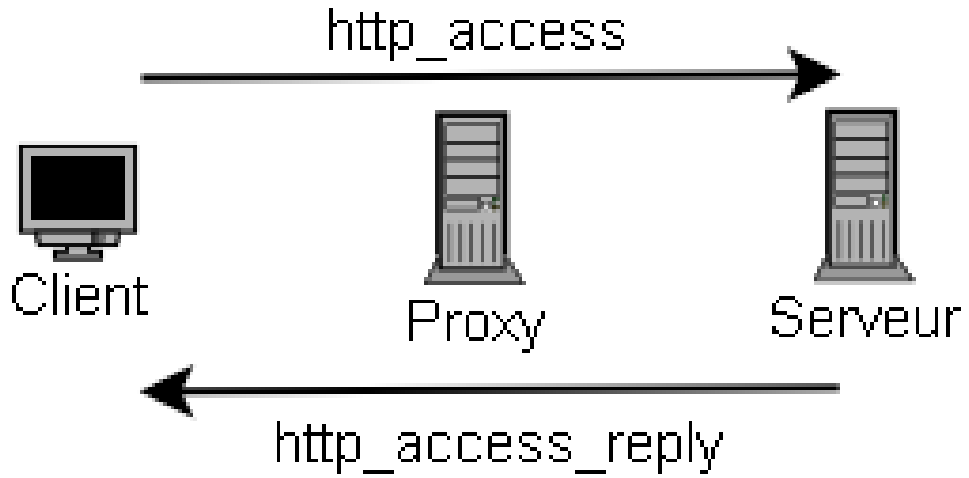
- Les ACL
- Syntaxe générale : acl name type value1 value2
- Principe : on introduit une acl par le mot clef acl, on lui donne un nom, un type et on en définit la ou les valeurs. Puis on l'associe à une action.
- Exemple :
  - acl LanInterne src 192.168.0.0/24
  - http_access allow LanInterne

Configuration

- ACL (suite)
- Principaux types :
  - src, dst : Adresses IP ou adresses de réseaux
  - srcdomain, dstdomain : noms de domaines
  - ports
  - method : acl Uploads method PUT POST
  - proto : HTTP, HTTPS, FTP, Gopher, etc.
    - acl WebSurfing method HTTP HTTPS
  - time : créneaux horaires
    - acl WorkingHours MTWHF 08:00-18:00

Configuration

- ACL (suite)
  - http_access : contrôle des requêtes émises par les clients du proxy
  - http_access_reply : contrôle des réponses aux requêtes des clients
  - no_cache : indique à Squid qu'il ne doit pas stocker les réponses dans son cache.



Configuration

- Exemples d'ACL
  - acl ALL src 0/0
  - acl MyNetwork 192.168.0.0/24
  - http_access allow MyNetwork
  - http_access deny ALL
- Prise en compte de l'heure
  - acl WorkingHours D 08:00-17:00
  - http_access deny !WorkingHours
  - http_access allow MyNetwork
  - http_access deny ALL

Configuration

- Exemple d'ACL sur les flux retour (http_access_reply)
- Blocage de certains fichiers par filtrage de type MIME
  - acl Movies rep_mime_type video/mpeg
  - acl Music rep_mime_type audio/mpeg
  - http_access_reply deny Movies
  - http_access_reply deny Muzic
  - http_access_reply allow All

Authentification

- Il est possible de rendre obligatoire l'authentification des utilisateurs par login/mot de passe
- Squid supporte nativement les méthodes suivantes :
  - BASIC
  - DIGEST
  - NTLM
- Il est possible d'utiliser des programmes externes pour étendre les capacités d'authentification
  - Exemple : winbindd pour utiliser Active Directory/Windows

Authentification

- Mise en oeuvre
  - Choix de la méthode (interne, externe)
  - Dans squid.conf :
    - acl AuthentUsers proxy_auth REQUIRED
    - http_access allow AuthentUsers
  - Astuces
    - Empêcher un utilisateur de partager son mot de passe
      - acl TooManyLogedUsers max_user_ip 2
      - http_access denay TooManyLogedUsers

Redirecteurs

- Les redirecteurs sont des programmes appelés par Squid pour effectuer des traitements sur les flux traités.
- Le filtrage d'URL est une tâche effectuée à l'aide de redirecteurs Squid.
  - SquidGuard
  - DansGuardian
- Le filtrage de contenu est également effectué à l'aide de redirecteurs.
  - SCAVR : permet de faire du filtrage antivirus sur les flux HTTP/FTP.

Redirecteurs

- Configuration dans squid.conf
- Déclaration du redirecteur (programme externe) :
  - redirect_program /path/to/program -options
- Nombre de process lancés en permanence
  - redirect_children 5
- ACL pour activer la redirection
  - redirect_access allow ALL
  - redirect_bypass on
    - Permet au flux de passer si aucun process n'est disponible.
    - redirect_bypass à off si le redirecteur doit assurer une fonction de sécurité !

SquidGuard

- Redirecteur Squid
- Filtrage d'URL : noms de domaines, mots-clefs, URL.
- Squid.conf :
  - redirect_program /path/to/squidGuard -c /path/to/squidGuard.conf
  - redirect_children 4

SquidGuard.conf

- logdir /usr/local/squidGuard/log
- dbhome /usr/local/squidGuard/db
- dest porn {
- domainlist porn/domains
- urllist porn/urls
- }
- acl {
- default {
- pass !porn all
- redirect http://localhost/cgi/blocked?clientaddr=%a&clientname=%n&clientuser=%i&clientgroup=%s&url=%u
- }
- }

ClamAV

Moteur antivirus OpenSource

Présentation

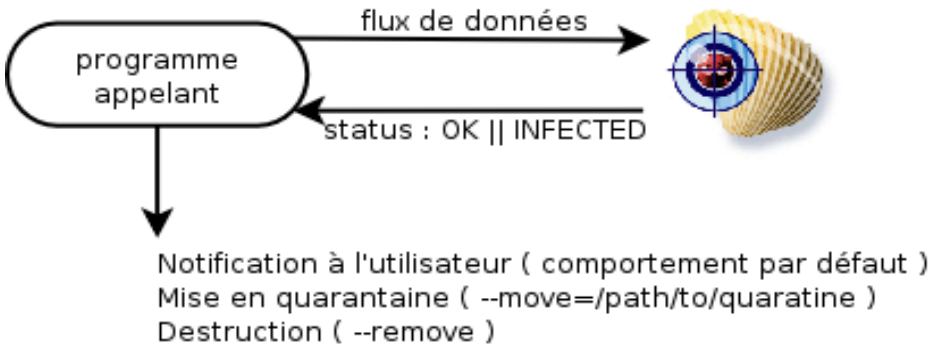
- ClamAV est un logiciel antivirus pour Unix (y compris Linux) distribué sous licence GPL.
- Initialement dédié à l'analyse des flux SMTP, ClamAV fournit un moteur d'analyse qui peut être utilisé en ligne de commande ou sous forme de démon.
- La librairie libclamav permet l'appel des fonctions d'analyse depuis des applications externes.
- Un effort important est mis sur la maintenance d'une base de signatures à jour.
  - 30.000 virus reconnus

Fonctionnalités

- Analyse de fichiers par recherche de signatures
- Support des formats d'archive et de compression courants :
  - zip, rar, tar, gzip, bzip2, MS Cabinet/CHM/SZDD
- Support des formats Portable Executable :
  - UPX, FSG, Petite
- Support des formats de BAL mailbox et Maildir
- Base de signatures généralement à jour, parfois la première (SoBig.I)

Architecture

- Principe de fonctionnement :
  - Les fichiers analysés sont passés au crible des signatures de la base locale
  - Le moteur retourne leur status ; OK ou INFECTED
  - Le programme appelant prend la décision.
    - Exemple : Clamscan : notification, mise en quarantaine ou destruction



Installation

- Pré-requis pour une installation à partir du code source
  - Librairies zlib
    - Si possible une version > 1.2.1
      - Contournement : --disable-zlib-vcheck
  - Librairies GNU MP 3
    - Vérification des bases de signatures
      - Contournement : --disable-dsig
  - Librairies bzip2
  - Utilisateur et groupe clamav
    - Sauf si installation dans un compte Utilisateur
      - ./configure –prefix=/home/joe –disable-clamav

Configuration (1)

- Deux fichiers :
  - [/etc/]clamd.conf :
    - Paramètres de configuration du démon clamd et des programmes associés
  - [/etc/]freshclam.conf :
    - Paramètres de configuration de l'utilitaire de mise à jour des bases de signatures.
- Par défaut, les fichiers créés ne sont pas utilisables :
  - Commenter la ligne `Example`

Configuration (2)

- clamd.conf
  - Journalisation
    - LogFile /path/to/fichier.log
    - LogSyslog
    - LogFileMaxSize (0 : nolimit)
    - LogTime
  - Connexion au démon
    - Socket
      - LocalSocket /path/to/clamav.sock
    - TCP
      - TCPSocket 3310 (valeur par défaut)

Configuration (3)

- clamd.conf (suite)
  - Gestion des ressources
    - MaxConnectionQueueLength
    - StreamMaxLength
    - MaxThreads
  - Sécurité
    - SelfCheck
    - ExitOnOOM
    - ArchiveMaxRecursion, ArchiveMaxFileSize, etc.
    - ArchiveBlockEncrypted

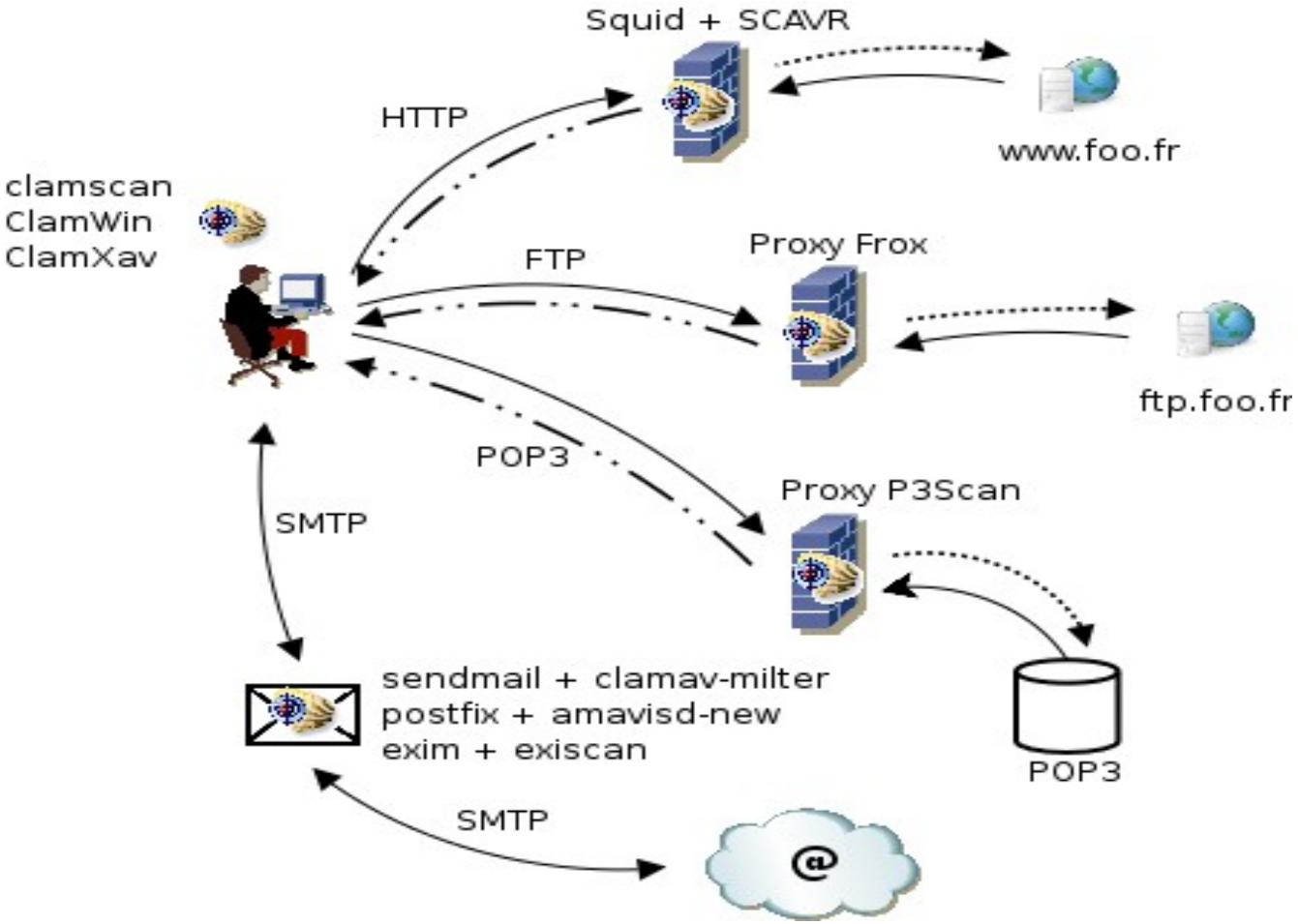
Mise à jour (1)

- Tâche essentielle : Pas de signature, pas de détection.
- Risques :
  - Absence de signature
    - Réactivité des *maintaners* ClamAV
      - Premier antivirus à fournir une signature pour le ver SoBig.I
    - Disponibilité des serveurs
      - Plusieurs mirroirs (round-robin DNS)
    - Intégrité des bases de signatures
      - Vérification des signatures (librairie GNU MP)

Mise à jour (2)

- Freshclam
  - Mode démon : freshclam -d
  - Crontab
  - Configuration : [/etc/]freshclam.conf
  - Enregistrement TXT sur current.cvd.clamav.net
  - current.cvd.clamav.net. 275 IN TXT "**0.83:29:752:1110216729**“
    - 0.83 : version de clamav
    - 29 : n° version de la base main.cvd
    - 725 : n° verrsion de la base dayly.cvd
    - 1110216729 : horodatage Epoch
  - Rechargement Clamd après mise à jour
    - NotifyClamd

Positionnement



Passerelle SMTP

- ClamAV / Sendmail
  - Utilisation du milter clamav-milter
    - Il est alors conseillé de désactiver le support de clamuko
      - ./configure –enable-milter –disable-clamuko
  - Configuration Sendmail
    - INPUT_MAIL_FILTER(`clmilter',`S=local:/var/run/clmilter.sock, F=, T=S:4m;R:4m')dnl define(`confINPUT_MAIL_FILTERS', `clmilter')
  - Configuration Clamd
    - /etc/clamd.conf : LocalSocket /var/run/clmilter.sock
  - Configuration clamav-milter
    - /usr/local/sbin/clamav-milter -lo /var/run/clmilter.sock

Références

- Projet Squid
  - http://www.squid-cache.org
- Projet SquidGuard
  - http://www.squidguard.org
- Projet ClamAV
  - http://www.clamav.net